Simulation Environment for Autonomous UAV

Eugénie Fontaine, Philip Giuliano, Christopher Gutierrez, Eric Osorio, and Nicolas Prulhiere
Dr. Mihhail Berezovski, Supervisor

EMBRY-RIDDLE Aeronautical University

Goal

To create a completely software-based UAV simulation by utilizing a digital, on-board computer called ArduCopter and a 3D game engine, Unity, to test the success rate of a mission for various UAV models. The goal is to display the "thought processes" of ArduCopter and the actual behavior of the simulated UAV model.

Input

- Altitude
- Latitude/Longitude
- Payload
- Motor Power

Problem

1. Being able to export the output data in a reliable manner, that unity can interpret.
2. How to record the output data. MAVProxy already gives real time updates, now we locate the data log.

Results

MAVProxy/ArduCopter and Unity:

- Receive commands
- Give output
- Unfortunately there is still no link between the two to work together

In the future:

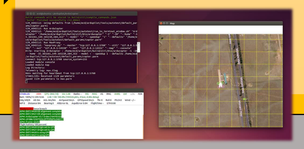
- Simulation will connect to ArduCopter through a Berkley network socket
- Acquiring funding for continuation of the project

Testing Environment

SITL

MAVProxy

- MAVProxy enables use of companion computing and multiple data links with ArduCopter
- Portable, command, and console based application that helps control SITL



Code

ArduCopter

Software capable of simulating advanced autonomous flight of quadrotor systems

Outputs

Output Data

Frame Data

Physical Model

- Unity physics creates the simulation
- Default physics is modified by adding code
  - Ex. Multi-directional drag (show below)

$$F_{Drag,N} = -C_{Drag,N} V_N^2 \frac{V_N}{|V_N|}$$

Outputs

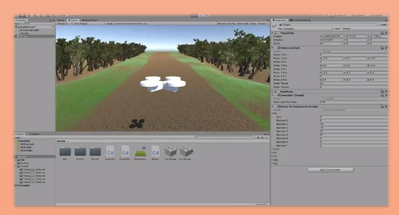
Physical model yields output so data can be analyzed and compared.
Expected outputs:

- Battery
- Motor Speed
- Altitude
- Distance Flown

Visualization

Simulation includes:

- Results of actuator signals
- Signals produced by various sensors
- 3D rendering of the mission



Unity

Game engine used to create a 3D physics simulation of the UAV. It interacts with ArduCopter and the team's customized code.